Advanced Search Engine Discovery

David Childers

License Declaration

Advanced Search Engine Discovery Application License

GNU General Public License

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see https://www.gnu.org/licenses/gpl-3.0.html#license-text.

Advanced Search Engine Discovery Application Documentation License:

Attribution-ShareAlike 4.0 International

You are free to:

Share - copy and redistribute the material in any medium or format for any purpose, even commercially.

Adapt - remix, transform, and build upon the material for any purpose, even commercially.

The licensor cannot revoke these freedoms as long as you follow the license terms.

Under the following terms:

Attribution - You must give appropriate credit , provide a link to the license, and indicate if changes were made . You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

ShareAlike - If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.

No additional restrictions - You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

Notices:

You do not have to comply with the license for elements of the material in the public domain or where your use is permitted by an applicable exception or limitation .

No warranties are given. The license may not give you all of the permissions necessary for your intended use. For example, other rights such as publicity, privacy, or moral rights may limit how you use the material.

https://creativecommons.org/licenses/by-sa/4.0/legalcode.en

In Memoriam

Dedicated To The Late Douglas Keith Foncree Jr.

A Master Of Computers, A Great Mentor, And A Wonderful Shipmate.

About

Imagination is the engine that drives creativity. It allows individuals to envision what does not yet exist, explore new possibilities, solve problems, express artistic visions, and empathize with others. Whether in science, business, art, or everyday problem-solving, imagination is essential for breaking free from the constraints of reality and fostering innovation and originality. Without imagination, creativity would be limited to existing knowledge and patterns, but with it, individuals are empowered to think beyond the present and shape the future.

Imaginatio Est Mater Creaturae

David Childers

September 2024

Executive Summary

This comprehensive solution equips users with advanced capabilities to excel in information discovery and apply their findings to achieve strategic goals.

Cover Image

Library Index Card Catalog

Index

Software / Documentation License Declaration

In Memoriam

About

Index

Introduction to Advanced Search Engine Discovery

Security Focused Web Browser

Open Source Security Based Web Browsers

Recommended Firefox Compatible Web Browser Extensions

Advanced Search Engine Discovery Application

Modifying The Advanced Search Engine Discovery HTML

Creating A Desktop Launcher

Introduction to Advanced Search Engine Discovery

Using multiple sources for information research and removing biases are critical strategies for ensuring the information's quality, credibility, and comprehensiveness. This can also improve the accuracy of findings and minimize the risk of using incorrect data.

Cross-Verification of Information: Using multiple sources allows the verification of information and ensures a more accurate and reliable representation of information, which prevents reliance on a single, potentially biased, or erroneous source.

Contextual Depth: Using multiple sources allows for a richer, more nuanced view of the subject. Researchers can consider all relevant factors and contexts, ensuring critical details are not overlooked.

Challenging Preconceived Notions: Using multiple sources forces users to engage with ideas that challenge their assumptions, foster objectivity, and help mitigate personal bias.

Diversifying Information Consumption: Search engines often provide personalized results based on past behavior, reinforcing confirmation bias by showing information aligned with a user's preferences. Using multiple sources prevents this.

Comparative Analysis and Critical Thinking: Using multiple sources sharpens the ability to compare, contrast, and critically evaluate the quality of information. This practice cultivates critical thinking and enhances the ability to assess the reliability of different types of data.

Varied Algorithms and Search Indexes: Using multiple search engines may reveal sources or perspectives a single engine could overlook. Each search engine uses unique algorithms and indexing processes, which can emphasize different factors, including relevance, popularity, or freshness.

Broader Coverage of Information: Using multiple engines increases the chances of finding more obscure or hard-to-access content, enriching the search results. No single search engine covers the entire web. Some engines have vast indexes, and others highlight niche or specialized sources.

Reduced Bias in Results: Search engines' ranking algorithms inherently reflect biases. Combining results from multiple engines helps mitigate these biases, offering a more balanced and objective view of the information.

Different Content Formats: Individual search engines emphasize different media formats. Using multiple platforms ensures access to information across various formats, such as articles, videos, podcasts, and infographics. This broadens the scope and depth of research.

Overcoming Regional Censorship: Some search engines are subject to regional restrictions, which may limit access to certain information. Using multiple engines, particularly those prioritizing privacy, helps bypass censorship and allows access to more content from unrestricted regions.

Protecting Privacy: Some search engines prioritize privacy, while others track user data. Alternating between engines prioritizing privacy offers protection while providing quality information, especially for sensitive topics.

Security Focused Web Browser

Using a security-focused web browser dedicated to web searches is becoming increasingly important in today's digital landscape. Such browsers allow users to configure their search experience for nonbiased results and protect privacy by anonymizing searches and block tracking. These features help ensure a more neutral, private, and secure online research experience.

Avoiding Personalization Bias: Traditional browsers and search engines often tailor results based on past behavior, preferences, and search history. This practice can create a bias by showing information aligned with previous interests, limiting exposure to new or contrasting viewpoints.

Bypassing Algorithmic Filters: Search engines and standard browsers often prioritize certain websites based on factors like advertising, popularity, or user engagement, which can inadvertently create bias by elevating certain viewpoints over others.

Blocking Tracking: Most conventional browsers allow websites and advertisers to track users' behavior across the web through cookies, creating detailed profiles of their interests and preferences. This leads to targeted advertising and biased search results that reflect prior behavior.

Preventing Search Engine Profiling: When users perform searches on standard browsers, search engines can profile them based on their search history, location, device, and other data points. These profiles are used to customize search results and deliver personalized ads, which can skew the information presented.

Protecting Sensitive Searches: Privacy-focused browsers are particularly valuable for users conducting sensitive searches, such as researching personal health, political issues, or financial matters. Without data anonymization, these searches could be tracked and stored, leading to intrusive ads, compromised privacy, or even data breaches.

Escaping Personalized Search Traps: Standard search engines often create "filter bubbles," where users are only exposed to information that aligns with their past behavior or preferences. This reduces exposure to new ideas and alternative viewpoints, reinforcing confirmation bias and limiting intellectual growth.

Accessing Unbiased Global Information: Many search engines use geo-targeting to present results based on the user's location, which can bias the information displayed.

User-Centric Configuration: Security-based browsers give users greater control over managing their data and how the browser interacts with websites. This allows users to configure the browser to meet their specific privacy needs, such as cookie and cache management and disabling JavaScript or other tracking mechanisms.

Third-Party Integrations for Privacy: Many security-focused browsers allow users to integrate privacy-enhancing extensions or plugins, such as ad-blockers, script blockers, or VPNs. These tools further help anonymize searches by preventing third-party advertisers and websites from tracking user activity.

Anonymizing to Avoid Surveillance: Governments and other entities may monitor online activities, including search queries, for surveillance purposes. Security add-ons can protect users from this surveillance by allowing the anonymization of searches and masking IP addresses, making it nearly impossible to trace searches back to the user. This protects both privacy and the ability to access unbiased, uncensored information without fear of surveillance.

Open Source Security Based Web Browsers

DEDICATED security-based SINGLE USE web browsers are essential in maintaining compartmentalized data security while providing unbiased and neutral search results. These web browsers must allow custom configuration capability and the ability to use additional browser-based extensions that provide data security and search integrity. No search engine login information should be used with these DEDICATED / SINGLE USE web browsers to maintain unbiased and neutral search results.

Waterfox is an open-source fork of the Firefox web browser with an emphasis on performance and privacy. Official Waterfox releases have been created for the Windows, macOS, Linux, FreeBSD, and Android platforms. Waterfox supports Firefox extensions and is also compatible with Google Chrome and Opera extensions. Waterfox disables telemetry and Pocket by default, but collects technical information about the user's device to properly update.

https://www.waterfox.net/

https://github.com/BrowserWorks/Waterfox

LibreWolf is an open-source fork of the Firefox browser with an emphasis on privacy and security. LibreWolf releases have been created for the Linux and FreeBSD platforms. LibreWolf supports Firefox extensions.

https://librewolf.net/

https://codeberg.org/librewolf/source

LibreWolf vs Waterfox

https://sourceforge.net/software/compare/LibreWolf-vs-Waterfox/

Recommended Firefox Compatible Web Browser Extensions

Testing different combinations of security extensions for each web browser for compatibility and performance optimization is essential. There is no guarantee that any particular combination performs better than other combinations.

Anonymize / Protect Web Searches With A Virtual Private Network

https://protonvpn.com/pricing
1 device - No logs - No ads - Unlimited & free forever

https://addons.mozilla.org/en-US/firefox/addon/proton-vpn-firefox-extension/
Browser GUI for ProtoVPN

Security Exploits

https://addons.mozilla.org/en-US/firefox/addon/noscript/
Protect against XSS other web security exploits.

Tracking

https://addons.mozilla.org/en-US/firefox/addon/disconnect/
Visualize and block the otherwise invisible websites that track your search and browsing history.

https://addons.mozilla.org/en-US/firefox/addon/ghostery/
Stop trackers, speed up websites and block ads everywhere.

https://addons.mozilla.org/en-US/firefox/addon/privacy-badger17/
Automatically learns to block invisible trackers.

https://addons.mozilla.org/en-US/firefox/addon/decentraleyes/
Protects you against tracking through "free", centralized, content delivery.

https://addons.mozilla.org/en-US/firefox/addon/clearurls/
Removes tracking elements from URLs.

Advertising Blocking

https://addons.mozilla.org/en-US/firefox/addon/ublock-origin/
A wide-spectrum content blocker with CPU and memory efficiency as a primary feature.

https://addons.mozilla.org/en-US/firefox/addon/tranquility-1/
Improve readability of web sites by removing unnecessary elements like ads, images, social widgets.

Cookies

https://addons.mozilla.org/en-US/firefox/addon/cookie-autodelete/
Control your cookies. When a tab closes, any cookies not being used are automatically deleted.

Browser Cache

https://addons.mozilla.org/en-US/firefox/addon/clearcache/
Clear browser cache with a single click or via the F9 key.

History Cleaner

https://addons.mozilla.org/en-US/firefox/addon/clear-browsing-data/
Delete browsing data directly from the browser toolbar. Clear cookies, history and cache.

https://addons.mozilla.org/en-US/firefox/addon/history-cleaner/
Deletes browsing history older than a specified number of days.

Advanced Search Engine Discovery Application

The following page contains the HTML code for the Advanced Search Engine Discovery website.

The code is designed to be used as a locally based HTML page.

Perform the following tasks:

Create a blank text file.

Open the blank text file with a text editor application.

Copy the Advanced Search Engine Discovery Application HTML code from this publication and paste it to the blank text file.

Save the file.

Rename the text file to search.htm.

Create an empty file folder on the desktop.

Rename the empty file folder to search.

Place the search.htm file into the search file folder.

```
<!-- HTML -->
```

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Advanced Search Engine Discovery</title>
  <style>
    /* Add some basic styling to the page */
    body {
      font-family: Arial, sans-serif;
    }
    .container {
      width: 80%;
      margin: 40px auto;
      text-align: center;
    }
    #searchQuery {
      width: 60%;
      padding: 10px;
      font-size: 16px;
    }
    button[type="submit"] {
      padding: 10px 20px;
      font-size: 16px;
      background-color: #4CAF50;
      color: #fff;
      border: none;
      border-radius: 5px;
      cursor: pointer;
    }
    button[type="submit"]:hover {
      background-color: #3e8e41;
    }
  </style>
</head>
<body>
  <main class="container">
    <h1>Advanced Search Engine Discovery</h1>
    <p>Multiple Search Engines Are Used To Perform Identical Information Inquiries And Compare The Results Simultaneously.<br><br></p>
    <p>The Following Search Engines Are Used:</p>
    <p><u>StartPage</u> - <u>Brave</u> - <u>Bing</u> - <u>Yahoo</u> - <u>Yandex</u> - <u>Presearch</u> - <u>DuckDuckGo</u>
  </p><br>

<style>
  #searchQuery {
    text-align: center;
  }
</style>

    <form id="searchForm">
      <input
        type="text"
        id="searchQuery"
        name="searchQuery"
        placeholder="Enter Query Topic To Be Searched"
        aria-label="Search Query Input"
      >
      <button type="submit">Search</button>
    </form>
  </main>

  <!-- JavaScript -->
  <script>
    (function() {
      const searchForm = document.getElementById('searchForm');
      const searchQueryInput = document.getElementById('searchQuery');

      searchForm.addEventListener('submit', (event) => {
        event.preventDefault();

        const query = searchQueryInput.value.trim();

        if (query === '') {
          alert('Please Enter Search Query Topic.');
          return;
        }

        const searchEngines = [
          { name: 'StartPage', url: 'https://www.startpage.com/do/dsearch?query=' },
          { name: 'Brave', url: 'https://search.brave.com/search?q=' },
          { name: 'Bing', url: 'https://www.bing.com/search?q=' },
          { name: 'Yahoo', url: 'https://search.yahoo.com/search?p=' },
          { name: 'Yandex', url: 'https://yandex.com/search/?text=' },
          { name: 'Presearch', url: 'https://presearch.com/search?q=' },
          { name: 'DuckDuckGo', url: 'https://duckduckgo.com/?q=' }
    ];

        searchEngines.forEach((engine) => {
          const searchUrl = `${engine.url}${encodeURIComponent(query)}`;
          window.open(searchUrl, '_blank');
        });
      });
    })();
  </script>

  <center>
    <p>* <b>PLEASE NOTE</b> * You Must Configure The Web Browser To Allow Multiple Pop-Ups.</p>
    <p>* <b>PLEASE NOTE</b> * Each Individual Search Engine Site Opens In A Different Browser Tab.</p>
  </center>

</body>
</html>
```

Modifying The Advanced Search Engine Discovery HTML

{ name: 'StartPage', url: 'https://www.startpage.com/do/dsearch?query=' },
{ name: 'Brave', url: 'https://search.brave.com/search?q=' },
{ name: 'Bing', url: 'https://www.bing.com/search?q=' },
{ name: 'Yahoo', url: 'https://search.yahoo.com/search?p=' },
{ name: 'Yandex', url: 'https://yandex.com/search/?text=' },
{ name: 'Presearch', url: 'https://presearch.com/search?q=' },
{ name: 'DuckDuckGo', url: 'https://duckduckgo.com/?q=' }

→ All search engines listed Should Have the search query instruction following the search engine URL.

→ Each search engine has specific search query instructions to use for query submission.

Examples of other search engine query submission instructions:

{ name: 'AOL Search', url: 'https://search.aol.com/aol/search?q=' },
{ name: 'Ask', url: 'https://www.ask.com/web?q=' },
{ name: 'Wikipedia', url: 'https://en.wikipedia.org/w/index.php?search=' },
{ name: 'Yippy', url: 'https://yippy.com/search?query=' },

→ The last listed search engine Should Not Have a comma after the parenthesis

}

{ name: 'DuckDuckGo', url: 'https://duckduckgo.com/?q=' }

→ All other search engines listed Should Have a comma placed after the parenthesis

},

{ name: 'Presearch', url: 'https://presearch.com/search?q=' },

Creating A Desktop Launcher

It is recommended that a desktop launcher is created to open the search.htm file contained in the search folder using the specified security-based web browser.

BSD (Net, Free, Open) - Linux - Solaris

To create a desktop shortcut for opening an HTML file with a specified web browser on BSD, Linux or Solaris, follow these steps:

Create a .desktop File:

- Open a text editor and create a new file named search.desktop.
- Place this file on your Desktop.

Edit the search.desktop File:

Copy the following into the search.desktop file.
  -Replace YOUR_BROWSER with the name of the preferred web browser (e.g., firefox, google-chrome, etc.)

```
----------------------
[Desktop Entry]
Version=1.0
Type=Application
Name=Open Search
Comment=Open search.htm with a web browser
Exec=YOUR_BROWSER ~/Desktop/search/search.htm
Icon=web-browser
Terminal=false
----------------------
```

Explanation of the fields:

- Version=1.0: Indicates the version of the .desktop file.
- Type=Application: Specifies that this is an application launcher.
- Name=Open Search: The name that will appear on the desktop shortcut.
- Comment=Open search.htm with a web browser: A brief description of the shortcut.
- Exec=YOUR_BROWSER ~/Desktop/search/search.htm: The command that will be executed.
    Replace YOUR_BROWSER with the desired browser, and ensure the path to the search.htm file is correct.
- Icon=web-browser: Sets the icon for the shortcut.
    You can use a custom icon by providing an absolute path to the image file.
- Terminal=false: Specifies that the program will not run in a terminal.

Make the .desktop File Executable:

Open a terminal and run the following command: $ chmod +x ~/Desktop/search.desktop

You will have a desktop shortcut named "Open Search" on your Desktop that opens the search.htm file with the specified web browser.

Windows

To create a desktop shortcut in Windows that opens an HTML file with a specified web browser, follow these steps:

Locate the search.htm File:

- Ensure the search.htm file is located in the search folder on your desktop (e.g., C:\Users\YourUsername\Desktop\search\search.htm).

Create a New Shortcut:

- Right-click on the desktop, and select New > Shortcut.

Set the Shortcut Target:

- Specify the path to the designated web browser followed by the path to the search.htm file in the "Type the location of the item" field. Use double quotes for paths that contain spaces.

For example: "C:\Program Files\Mozilla Firefox\firefox.exe" "C:\Users\YourUsername\Desktop\search\search.htm"

Replace "C:\Program Files\Mozilla Firefox\firefox.exe" with the actual path to your preferred web browser executable (e.g., Firefox, Chrome, or Edge) and YourUsername with your Windows username.

Name the Shortcut:

- Give your shortcut a meaningful name, such as "Open Search".

Finish Creating the Shortcut:

- Click Finish to create the shortcut.

Optional: Change the Shortcut Icon:

- Right-click the new shortcut, select Properties, and then click Change Icon to set a custom icon if desired.

Now, double-clicking the "Open Search" shortcut on your desktop will open the search.htm file in the specified web browser.

macOS

To create a desktop shortcut on macOS that opens an HTML file with a specified web browser, follow these steps:

Locate the search.htm File:

- Ensure the search.htm file is located in the search folder on your Desktop (e.g., ~/Desktop/search/search.htm).

Create an AppleScript:

- Open the Script Editor app (located in Applications > Utilities).

- Write an AppleScript to open the search.htm file with your preferred web browser. Replace YOUR_BROWSER with the name of the browser (e.g., Google Chrome, Firefox, or Safari):

```
----------------------
  tell application "YOUR_BROWSER"
      open POSIX file "/Users/YourUsername/Desktop/search/search.htm"
      activate
  end tell
----------------------
```

- Make sure to replace YourUsername with your actual macOS username and YOUR_BROWSER with the desired browser.

Save the AppleScript as an Application:

- Go to File > Export.

- Set File Format to "Application".

- Name the file (e.g., "Open Search") and save it to your Desktop.

Change the Icon (Optional):

- You can change the application’s icon by copying an image and pasting it into the icon preview in the Get Info window for the newly created application.

Now, you have a desktop shortcut named "Open Search" that will open the search.htm file in the specified web browser. Double-clicking this application will launch the web browser and display the HTML file.